

```

runner% cat eval1.c
/* arith.c: simple parser -- bold evaluates */
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <math.h> /* for pow */
char next;
double E(void);
double T(void);
double S(void);
double F(void);
void error(void);
void scan(void);
double evaluate(double arg1, char op, double arg2);

void main(void)
{
    double res;
    scan();
    res = E();
    if (next != '#') error();
    else
        printf("Result is: %.5f\n", res);
}

double E(void)
{
    char save;
    double res, arg1, arg2;
    res = T();
    while (next == '+' || next == '-') {
        save = next;
        arg1 = res;
        scan();
        arg2 = T();
        res = evaluate(arg1, save, arg2);
    }
    return res;
}

double T(void)
{
    char save;
    double res;
    double save;
    while (next == '(' || next == ')') {
        save = next;
        arg1 = res;
        scan();
        arg2 = T();
        res = evaluate(arg1, save, arg2);
    }
    return res;
}

double S(void)
{
    char save;
    double res, arg1, arg2;
    res = F();
    if (next == '^') {
        save = next;
        arg1 = res;
        scan();
        arg2 = S();
        res = evaluate(arg1, save, arg2);
    }
    return res;
}

double F(void)
{
    char save;
    double res;
    if (isdigit(next)) {
        save = next;
        scan();
        return (double)(save - '0');
    }
    else if (next == '(') {
        scan();
        res = E();
        if (next == ')') scan();
        else error();
        return res;
    }
    else {
        error();
        return 0;
    }
}

```

